Weighted Nuclear Norm Minimization with Application to Image Denoising
— Source link ⧉

Shuhang Gu, Lei Zhang, Wangmeng Zuo, Xiangchu Feng

Institutions: Harbin Institute of Technology, Xidian University

Related papers:

- Image Denoising by Sparse 3-D Transform-Domain Collaborative Filtering

- Beyond a Gaussian Denoiser: Residual Learning of Deep CNN for Image Denoising

- A non-local algorithm for image denoising

- Image Denoising Via Sparse and Redundant Representations Over Learned Dictionaries

- Non-local sparse models for image restoration

Weighted Nuclear Norm Minimization with Application to Image Denoising

Shuhang Gu[1], Lei Zhang[1], Wangmeng Zuo[2], Xiangchu Feng[3]

[1]Dept. of Computing, The Hong Kong Polytechnic University, Hong Kong, China

[2]School of Computer Science and Technology, Harbin Institute of Technology, Harbin, China

[3]Dept. of Applied Mathematics, Xidian University, Xi'an, China

{cssgu, cslzhang}@comp.polyu.edu.hk, cswmzuo@gmail.com, xcfeng@mail.xidian.edu.cn

Abstract

As a convex relaxation of the low rank matrix factorization problem, the nuclear norm minimization has been attracting significant research interest in recent years. The standard nuclear norm minimization regularizes each singular value equally to pursue the convexity of the objective function. However, this greatly restricts its capability and flexibility in dealing with many practical problems (e.g., denoising), where the singular values have clear physical meanings and should be treated differently. In this paper we study the weighted nuclear norm minimization (WNNM) problem, where the singular values are assigned different weights. The solutions of the WNNM problem are analyzed under different weighting conditions. We then apply the proposed WNNM algorithm to image denoising by exploiting the image nonlocal self-similarity. Experimental results clearly show that the proposed WNNM algorithm outperforms many state-of-the-art denoising algorithms such as BM3D in terms of both quantitative measure and visual perception quality.

1. Introduction

Low rank matrix approximation, which aims to recover the underlying low rank matrix from its degraded observation, has a wide range of applications in computer vision and machine learning. For instance, the low rank nature of matrix formed by human facial images allows us to reconstruct the occluded/corrupted faces [8, 20, 30]. The Netflix customer data matrix is believed to be low rank due to the fact that the customers' choices are mostly affected by a few common factors [24]. The video clip captured by a static camera has a clear low rank property, based on which background modeling and foreground extraction can be conducted [27, 23]. It is also shown that the matrix formed by nonlocal similar patches in a natural image is of low rank, which can be exploited for high performance image restoration tasks [26]. Owe to the rapid development of convex and non-convex optimization techniques, in recent years there are a flurry of studies in low rank matrix approximation, and many important models and algorithms have been reported [25, 2, 16, 13, 14, 4, 27, 3, 20, 19, 21, 11].

Low rank matrix approximation methods can be generally categorized into two categories: the low rank matrix factorization (LRMF) methods [25, 2, 16, 13] and the nuclear norm minimization (NNM) methods [14, 4, 27, 3, 20, 19, 21, 11]. Given a matrix Y, LRMF aims to find a matrix X, which is as close to Y as possible under certain data fidelity functions, while being able to be factorized into the product of two low rank matrices. A variety of LRMF methods have been proposed, ranging from the classical singular value decomposition (SVD) to the many L_1-norm robust LRMF algorithms [25, 2, 16, 13].

The LRMF problem is basically a nonconvex optimization problem. Another line of research for low rank matrix approximation is NNM. The nuclear norm of a matrix X, denoted by $\|X\|_*$, is defined as the sum of its singular values, i.e., $\|X\|_* = \sum_i |\sigma_i(X)|_1$, where $\sigma_i(X)$ means the i-th singular value of X. NNM aims to approximate Y by X, while minimizing the nuclear norm of X. One distinct advantage of NNM lies in that it is the tightest convex relaxation to the non-convex LRMF problem with certain data fidelity term, and hence it has been attracting great research interest in recent years. On one side, Candes and Recht [6] proved that most low rank matrices can be perfectly recovered by solving an NNM problem; on the other side, Cai et al. [3] proved that the NNM based low rank matrix approximation problem with F-norm data fidelity can be easily solved by a soft-thresholding operation on the singular values of observation matrix. That is, the solution of

$$\hat{X} = \arg\min_X \|Y - X\|_F^2 + \lambda\|X\|_*, \qquad (1)$$

where λ is a positive constant, can be obtained by

$$\hat{X} = US_\lambda(\Sigma)V^T, \qquad (2)$$

where $Y = U\Sigma V^T$ is the SVD of Y and $S_\lambda(\Sigma)$ is the soft-thresholding function on diagonal matrix Σ with parameter

λ. For each diagonal element $\boldsymbol{\Sigma}_{ii}$ in $\boldsymbol{\Sigma}$, there is

$$\mathcal{S}_\lambda(\boldsymbol{\Sigma})_{ii} = \max(\boldsymbol{\Sigma}_{ii} - \lambda, 0). \qquad (3)$$

The above singular value soft-thresholding method has been widely adopted to solve many NNM based problems, such as matrix completion [6, 5, 3], robust principle component analyze (RPCA) [4, 27], low rank textures [29] and low rank representation (LRR) for subspace clustering [20].

Although NNM has been widely used for low rank matrix approximation, it still has some problems. In order to pursue the convex property, the standard nuclear norm treats each singular value equally, and as a result, the soft-thresholding operator in (3) shrinks each singular value with the same amount λ. This, however, ignores the prior knowledge we often have on the matrix singular values. For instance, the column (or row) vectors in the matrix often lie in a low dimensional subspace; the larger singular values are generally associated with the major projection orientations, and thus they'd better be shrunk less to preserve the major data components. Clearly, NNM and its corresponding soft-thresholding operator fail to take advantage of such prior knowledge. Though the model in (1) is convex, it is not flexible enough to deal with many real problems. Zhang et al. proposed a Truncated Nuclear Norm Regularization (TNNR) method [28]. However, TNNR is not flexible enough since it makes a binary decision that whether to regularize a specific singular value or not.

To improve the flexibility of nuclear norm, we propose the weighted nuclear norm and study its minimization. The weighted nuclear norm of a matrix $\boldsymbol{X}$ is defined as

$$\|\boldsymbol{X}\|_{\boldsymbol{w},*} = \sum_i |w_i \sigma_i(\boldsymbol{X})|_1, \qquad (4)$$

where $\boldsymbol{w} = [w_1, \ldots, w_n]$ and $w_i \geq 0$ is a non-negative weight assigned to $\sigma_i(\boldsymbol{X})$. The weighted nuclear norm minimization (WNNM) is not convex in general case, and it is more difficult to solve than NNM. So far little work has been reported on the WNNM problem.

In this paper, we study in detail the WNNM problem with F-norm data fidelity. The solutions under different weight conditions are analyzed, and the proposed algorithm of WNNM is as efficient as that of the NNM problem. WNNM generalizes NNM, and it greatly improves the flexibility of NNM. Different weights or weighting rules can be introduced based on the prior knowledge and understanding of the problem, and WNNM will benefit the estimation of the latent data in return.

As an important application, we adopt the proposed WNNM algorithm to image denoising. The goal of image denoising is to estimate the latent clean image from its noisy observation. As a classical and fundamental problem in low level vision, image denoising has been extensively studied for many years; however, it is still an active

research topic because denoising is an ideal test bed to investigate and evaluate the statistical image modeling techniques. In recent years, the exploitation of image nonlocal self-similarity (NSS) has boosted significantly the image denoising performance [1, 7, 10, 22, 12, 9]. The NSS prior refers to the fact that for a given local patch in a natural image, one can find many similar patches to it across the image. The benchmark BM3D [7] algorithm and the state-of-the-art algorithms such as LSSC [22] and NCSR [10] are all based on the NSS prior. Intuitively, by stacking the nonlocal similar patch vector into a matrix, this matrix should be a low rank matrix and has sparse singular values. This assumption is validated by Wang *et al.* in [26], where they called it the nonlocal spectral prior. Therefore, the low rank matrix approximation method can be used to design denoising algorithms. The NNM method was adopted in [15] for video denoising. In [9], Dong et al. combined NNM and $L_{2,1}$-norm group sparsity for image restoration, and demonstrated very competitive results.

The contribution of this paper is two-fold. First, we analyze in detail the WNNM optimization problem and provide the solutions under different weight conditions. Second, we adopt the proposed WNNM algorithm to image denoising to demonstrate its great potentials in low level vision applications. The experimental results showed that WNNM outperforms state-of-the-art denoising algorithms not only in PSNR index, but also in local structure preservation, leading to visually more pleasant denoising outputs.

2. Low-Rank Minimization with Weighted Nuclear Norm

2.1. The Problem

As reviewed in Section 1, low rank matrix approximation can be achieved by low rank matrix factorization and nuclear norm minimization (NNM), while the latter can be a convex optimization problem. NNM is getting increasingly popular in recent years because it is proved in [6] that most low rank matrices can be well recovered by NNM, and it is shown in [3] that NNM can be efficiently solved. More specifically, by using the F-norm to measure the difference between observed data matrix $\boldsymbol{Y}$ and the latent data matrix $\boldsymbol{X}$, the NNM model in (1) has an analytical solution (refer to (2)) via the soft-thresholding of singular values (refer to (3)). NNM penalizes the singular values of $\boldsymbol{X}$ equally. Thus, the same soft-threshold (i.e., λ) will be applied to all the singular values, as shown in (3). This is not very reasonable since different singular values may have different importance and hence they should be treated differently. To this end, we use the weighted nuclear norm defined in (4) to regularize $\boldsymbol{X}$, and propose the following weighted nuclear norm minimization (WNNM) problem

$$\min_{\boldsymbol{X}} \|\boldsymbol{Y} - \boldsymbol{X}\|_F^2 + \|\boldsymbol{X}\|_{\boldsymbol{w},*}. \qquad (5)$$

The WNNM problem, however, is much more difficult to optimize than NNM since the objective function in (5) is not convex in general. In [3], the sub-gradient method is employed to derive the solution of NNM; unfortunately, similar derivation cannot be applied to WNNM since the sub-gradient conditions are no longer satisfied. In subsection 2.2, we will discuss the solution of WNNM in detail. Obviously, NNM is a special case of WNNM when all the weights $w_{i=1...n}$ are the same. Our solution will cover the solution of NNM in [3], while our derivation is much simpler than the complex sub-gradient based derivation in [3].

2.2. Optimization

Before analyzing the optimization of WNNM, we first give following three lemmas.

Lemma 1. $\forall A, B \in \Re^{m \times n}$ that satisfy $A^T B = 0$, we have

$$(1) \|A + B\|_{w,*} \geq \|A\|_{w,*};$$
$$(2) \|A + B\|_F \geq \|A\|_F.$$

Lemma 2. $\forall M = \begin{bmatrix} A & B \\ C & D \end{bmatrix}$ with $A \in \Re^{m \times m}$ and $D \in \Re^{n \times n}$, if the weights satisfy $w_1 \geq \cdots \geq w_{m+n} \geq 0$, we have

$$\|M\|_{w,*} \geq \|A\|_{w_1,*} + \|D\|_{w_2,*},$$

where $w = [w_1, \ldots, w_{m+n}]$, $w_1 = [w_1, \ldots, w_m]$ and $w_2 = [w_{m+1}, \ldots, w_{m+n}]$.

Lemma 3. $\forall A \in \Re^{n \times n}$ and a diagonal non-negative matrix $W \in \Re^{n \times n}$ with non-ascending ordered diagonal elements, let $A = X\Phi Y^T$ be the SVD of A, we have

$$\sum_i \sigma_i(A)\sigma_i(W) = \max_{U^T U = I, V^T V = I} tr[WU^T AV],$$

where I is the identity matrix, $\sigma_i(A)$ and $\sigma_i(W)$ are the i-th singular values of matrices A and W, respectively. When $U = X$ and $V = Y$, $tr[WU^T AV]$ reaches its maximum value.

The proofs of the above lemmas can be found in the supplementary material. We then have the following theorem, which guarantees that the column and row spaces of the solution to the WNNM problem in (5) still lie in the corresponding spaces of the observation data matrix Y.

Theorem 1. $\forall Y \in \Re^{m \times n}$, denote by $Y = U\Sigma V^T$ the SVD of it. For the WNNM problem in (5) with non-negative weight vector w, its solution $\hat{X}$ can be written as $\hat{X} = U\hat{B}V^T$, where $\hat{B}$ is the solution of the following optimization problem

$$\hat{B} = \arg\min_B \|\Sigma - B\|_F^2 + \|B\|_{w,*}. \quad (6)$$

Proof. Denote by $U_\perp$ the set of orthogonal bases of the complementary space of U, we can write X as $X = UA_1 +$ $U_\perp A_2$, where A_1 and A_2 are the components of X in subspaces U and $U_\perp$, respectively. Then we have

$$
\begin{aligned}
f(X) &= \|Y - X\|_F^2 + \|X\|_{w,*} \\
&= \|U\Sigma V^T - UA_1 - U_\perp A_2\|_F^2 + \|UA_1 + U_\perp A_2\|_{w,*} \\
&\geq \|U\Sigma V^T - UA_1\|_F^2 + \|UA_1\|_{w,*} \quad (Lemma \quad 1).
\end{aligned}
$$

Similarly, for the row space bases V, we have

$$f(X) \geq \|U\Sigma V^T - UBV^T\|_F^2 + \lambda\|UBV^T\|_{w,*}.$$

Orthonormal matrices U and V will not change the F-norm and weighted nuclear norm, and thus we have

$$f(X) \geq \|\Sigma - B\|_F^2 + \lambda\|B\|_{w,*}.$$

Therefore, if we have the solution of the minimization problem in (6), the solution of the original WNNM problem in (5) can be obtained as $\hat{X} = U\hat{B}V^T$. $\qquad \square$

Based on the above lemmas and theorem, we discuss the solution of the WNNM problem under three situations: the weights $w_{i=1\cdots n}$ are in a non-ascending order, in an arbitrary order, and in a non-descending order, respectively.

2.2.1 The weights are in a non-ascending order

Based on Theorem 1, we have the globally optimal solution of the WNNM problem in (5) in the case that $w_1 \geq \cdots \geq w_n \geq 0$. We have the following theorem.

Theorem 2. *If weights satisfy* $w_1 \geq \cdots \geq w_n \geq 0$, *the WNNM problem in* (5) *has a globally optimal solution*

$$\hat{X} = US_w(\Sigma)V^T,$$

where $Y = U\Sigma V^T$ *is the SVD of* Y, *and* $S_w(\Sigma)$ *is the generalized soft-thresholding operator with weight vector* w

$$S_w(\Sigma)_{ii} = \max(\Sigma_{ii} - w_i, 0).$$

Proof. Considering the optimization problem in (6), and assuming that Λ_B is a diagonal matrix which has the same diagonal elements as matrix B, we have

$$
\begin{aligned}
&\|\Sigma - B\|_F^2 + \lambda\|B\|_{w,*} \\
=&\|\Sigma - \Lambda_B - (B - \Lambda_B)\|_F^2 + \|\Lambda_B + (B - \Lambda_B)\|_{w,*} \\
\geq&|\Sigma - \Lambda_B\|_F^2 + \|\Lambda_B\|_{w,*} \qquad (Lemma \quad 2).
\end{aligned}
$$

Thus, in such a weight condition, the optimal solution of (6) has a diagonal form Λ_B. Both Σ and Λ_B are diagonal matrices, and the solution can be obtained by soft-thresholding operation on each element. Based on the conclusion in Theorem 1, the optimal solution of (5) is $\hat{X} = US_w(\Sigma)V^T$. $\quad \square$

Theorem 2 greatly extends the Theorem 2.1 in [3] (which is described by (1)-(3) in this paper). We show that if the weights $w_{i=1...n}$ are in a non-ascending order, not necessarily have the same value, the WNNM problem is still convex and the optimal solution can still be obtained by soft-thresholding on the singular values but with different thresholds. The Theorem 2.1 given by Cai *et al.* in [3] is a special case of our Theorem 2. Compared with the complex sub-gradient based proof in [3], however, our proof is much more concise.

2.2.2 The weights are in an arbitrary order

In the case that weights $w_{i=1...n}$ are not in a non-ascending order but in an arbitrary order, the WNNM problem in (5) is non-convex, and thus we cannot have a global minimum of it. We propose an iterative algorithm to solve it.

In Theorem 1, we have proved that the solution of (5) can be obtained by solving (6). Let $\boldsymbol{B} = \boldsymbol{P}\boldsymbol{\Lambda}\boldsymbol{Q}^T$ be the SVD of $\boldsymbol{B}$. We solve the following optimization problem iteratively

$$(\hat{\boldsymbol{P}}, \hat{\boldsymbol{\Lambda}}, \hat{\boldsymbol{Q}}) = \arg\min_{\boldsymbol{P}, \boldsymbol{\Lambda}, \boldsymbol{Q}} \|\boldsymbol{P}\boldsymbol{\Lambda}\boldsymbol{Q}^T - \boldsymbol{\Sigma}\|_F^2 + \|\boldsymbol{P}\boldsymbol{\Lambda}\boldsymbol{Q}^T\|_{\boldsymbol{w},*},$$
$$s.t. \boldsymbol{P}^T\boldsymbol{P} = \boldsymbol{I}, \boldsymbol{Q}^T\boldsymbol{Q} = \boldsymbol{I} \tag{7}$$

where $\boldsymbol{I}$ is the identity matrix.

Step1: Given non-negative diagonal matrix $\boldsymbol{\Lambda}$, we solve

$$(\hat{\boldsymbol{P}}, \hat{\boldsymbol{Q}}) = \arg\min_{\boldsymbol{P}, \boldsymbol{Q}} \|\boldsymbol{P}\boldsymbol{\Lambda}\boldsymbol{Q}^T - \boldsymbol{\Sigma}\|_F^2.$$

Based on the definition of Frobenius norm, we have

$$\begin{aligned} &\min_{\boldsymbol{P}, \boldsymbol{Q}} \|\boldsymbol{P}\boldsymbol{\Lambda}\boldsymbol{Q}^T - \boldsymbol{\Sigma}\|_F^2 \\ =&\min_{\boldsymbol{P}, \boldsymbol{Q}} tr[(\boldsymbol{P}\boldsymbol{\Lambda}\boldsymbol{Q}^T - \boldsymbol{\Sigma})(\boldsymbol{P}\boldsymbol{\Lambda}\boldsymbol{Q}^T - \boldsymbol{\Sigma})^T] \\ =&tr[\boldsymbol{\Lambda}\boldsymbol{\Lambda} + \boldsymbol{\Sigma}\boldsymbol{\Sigma}] - 2\max_{\boldsymbol{P}, \boldsymbol{Q}} tr[\boldsymbol{P}\boldsymbol{\Lambda}\boldsymbol{Q}^T\boldsymbol{\Sigma}^T] \\ =&tr[\boldsymbol{\Lambda}\boldsymbol{\Lambda} + \boldsymbol{\Sigma}\boldsymbol{\Sigma}] - 2\sum_i \sigma_i(\boldsymbol{\Sigma})\sigma_i(\boldsymbol{\Lambda}) \quad (Lemma \quad 3) \end{aligned}$$

and the optimal solution of $\boldsymbol{P}$ and $\boldsymbol{Q}$ are the column and row bases of the SVD of matrix $\boldsymbol{\Lambda}$. As $\boldsymbol{\Lambda}$ is already a diagonal matrix, $\boldsymbol{P}$ and $\boldsymbol{Q}$ are permutation matrices which make the diagonal matrix $\boldsymbol{P}\boldsymbol{\Lambda}\boldsymbol{Q}^T$ have non-ascending ordered diagonal elements.

Step2: Given orthogonal matrices $\boldsymbol{P}$ and $\boldsymbol{Q}$, we solve

$$\hat{\boldsymbol{\Lambda}} = \arg\min_{\boldsymbol{\Lambda}} \|\boldsymbol{P}\boldsymbol{\Lambda}\boldsymbol{Q}^T - \boldsymbol{\Sigma}\|_F^2 + \|\boldsymbol{P}\boldsymbol{\Lambda}\boldsymbol{Q}^T\|_{\boldsymbol{w},*}.$$

Since $\boldsymbol{P}\boldsymbol{\Lambda}\boldsymbol{Q}^T$ is a diagonal matrix which has non-ascending ordered elements, we have

$$\hat{\boldsymbol{\Lambda}} = \arg\min_{\boldsymbol{\Lambda}} \sum_i \|(\boldsymbol{P}\boldsymbol{\Lambda}\boldsymbol{Q}^T)_{ii} - \boldsymbol{\Sigma}_{ii}\|_2^2 + |w_i \cdot (\boldsymbol{P}\boldsymbol{\Lambda}\boldsymbol{Q}^T)_{ii}|_1.$$

The soft-thresholding operation can be performed on each element of diagonal matrix $\boldsymbol{P}\boldsymbol{\Lambda}\boldsymbol{Q}^T$. Because $\boldsymbol{P}$ and $\boldsymbol{Q}$ are permutation matrices which only change the positions of diagonal elements, we have

$$\hat{\boldsymbol{\Lambda}} = \boldsymbol{P}^T \mathcal{S}_{\boldsymbol{w}}(\boldsymbol{\Sigma})\boldsymbol{Q}.$$

By iterating between the above two steps, (6) can be solved iteratively via sorting the diagonal elements and shrinking the singular values:

$$\begin{cases} (\boldsymbol{P}_{(k+1)}^T, \boldsymbol{\Phi}, \boldsymbol{Q}_{(k+1)}^T) = SVD(\boldsymbol{\Lambda}_{(k)}); \\ \boldsymbol{\Lambda}_{(k+1)} = \boldsymbol{P}_{(k+1)}^T \mathcal{S}_{\boldsymbol{w}}(\boldsymbol{\Sigma})\boldsymbol{Q}_{(k+1)}. \end{cases} \tag{8}$$

Based on the conclusion of Theorem 1, the final estimation of $\hat{\boldsymbol{X}}$ can be obtained by

$$\hat{\boldsymbol{X}} = \boldsymbol{U}\hat{\boldsymbol{P}}^T \mathcal{S}_{\boldsymbol{w}}(\boldsymbol{\Sigma})\hat{\boldsymbol{Q}}\boldsymbol{V}^T.$$

2.2.3 The weights are in a non-descending order

At last, we consider another special but very useful case, i.e., the weights $w_{i,...,n}$ are in a non-descending order. Based on the iterative algorithm proposed in subsection 2.2.2, we have the following corollary.

Corollary 1. *If the weights satisfy* $0 \leq w_1 \leq \ldots \leq w_n$*, the iterative algorithm described in subsection 2.2.2 will have a fixed point* $\hat{\boldsymbol{X}} = \boldsymbol{U}\mathcal{S}_{\boldsymbol{w}}(\boldsymbol{\Sigma})\boldsymbol{V}^T$*.*

Proof. In (8), by initializing $\boldsymbol{\Lambda}_{(0)}$ as any diagonal matrix with non-ascending ordered diagonal elements, we have

$$\begin{cases} (\boldsymbol{P}_{(1)} = \boldsymbol{I}, \boldsymbol{\Phi} = \boldsymbol{\Lambda}_{(0)}, \boldsymbol{Q}_{(1)} = \boldsymbol{I}) = SVD(\boldsymbol{\Lambda}_{(0)}); \\ \boldsymbol{\Lambda}_{(1)} = \boldsymbol{I}\mathcal{S}_{\boldsymbol{w}}(\boldsymbol{\Sigma})\boldsymbol{I} = \mathcal{S}_{\boldsymbol{w}}(\boldsymbol{\Sigma}). \end{cases}$$

Consequently, $\forall 0 < i < j \leq n$, we have $\boldsymbol{\Sigma}_{ii} \geq \boldsymbol{\Sigma}_{jj}$ and $w_i \leq w_j$. After soft-thresholding operation, $\boldsymbol{\Lambda}_{(1)} = \mathcal{S}_{\boldsymbol{w}}(\boldsymbol{\Sigma})$ still satisfies the non-ascending order. Thus in the next iteration, $\boldsymbol{P}$ and $\boldsymbol{Q}$ are still identity matrices, and the optimization of (7) reaches a fix point. Based on the conclusion of Theorem 1, we obtain a fix point estimation of $\boldsymbol{X}$ by $\hat{\boldsymbol{X}} = \boldsymbol{U}\mathcal{S}_{\boldsymbol{w}}(\boldsymbol{\Sigma})\boldsymbol{V}^T$. $\square$

The conclusion in Corollary 1 is very important and useful. The singular values of a matrix are always sorted in a non-ascending order, and the larger singular values usually correspond to the subspaces of more important components of the data matrix. Therefore, we'd better shrink the larger singular values less, that is, assigning smaller weights to the larger singular values in the weighted nuclear norm. In such a case, Corollary 1 guarantees that our proposed iterative algorithm has a fixed point. Furthermore, this fixed point has an analytical form (i.e., $\hat{\boldsymbol{X}} = \boldsymbol{U}\mathcal{S}_{\boldsymbol{w}}(\boldsymbol{\Sigma})\boldsymbol{V}^T$). Hence, in practice we do not need to really iterate, but directly get the desired solution in a single step, which makes the proposed method very efficient. As we will see in the following Section 3, Corollary 1 offers us an effective denoising algorithm, which shows superior denoising performance to almost all state-of-the-art denoising algorithms we can find.

4

3. WNNM for Image Denoising

Image denoising aims to reconstruct the original image x from its noisy observation $y = x + n$, where n is assumed to be additive Gaussian white noise with zero mean and variance σ_n^2. Denoising is not only an important pre-processing step for many vision applications, but also an ideal test bed for evaluating statistical image modeling methods. The seminal work of nonlocal means [1] triggers the wide study of nonlocal self-similarity (NSS) based methods for image denoising. NSS refers to the fact that there are many repeated local patterns across a natural image, and those nonlocal similar patches to a given patch can help much the reconstruction of it. The NSS based image denoising algorithms such as BM3D [7], LSSC [22] and NCSR [10] have achieved state-of-the-art denoising results.

For a local patch y_j in image y, we can search for its nonlocal similar patches across the image (in practice, in a large enough local window) by methods such as block matching [7]. By stacking those nonlocal similar patches into a matrix, denote by Y_j, we have $Y_j = X_j + N_j$, where X_j and N_j are the patch matrices of original image and noise, respectively. Intuitively, X_j should be a low rank matrix, and the low rank matrix approximation methods can be used to estimate X_j from Y_j. By aggregating all the denoised patches, the whole image can be estimated. Indeed, the NNM method has been adopted in [15] for video denoising. We apply the proposed WNNM model to Y_j to estimate X_j for image denoising. By using the noise variance σ_n^2 to normalize the F-norm data fidelity term $\|Y_j - X_j\|_F^2$, we have the following energy function:

$$\hat{X}_j = \arg\min_{X_j} \frac{1}{\sigma_n^2} \|Y_j - X_j\|_F^2 + \|X_j\|_{w,*}. \quad (9)$$

Obviously, the key issue now is the determination of the weight vector w. For natural images, we have the general prior knowledge that the larger singular values of X_j are more important than the smaller ones since they represent the energy of the major components of X_j. In the application of denoising, the larger the singular values, the less they should be shrunk. Therefore, a natural idea is that the weight assigned to $\sigma_i(X_j)$, the i-th singular value of X_j, should be inversely proportional to $\sigma_i(X_j)$. We let

$$w_i = c\sqrt{n} \Big/ (\sigma_i(X_j) + \varepsilon), \quad (10)$$

where $c > 0$ is a constant, n is the number of similar patches in Y_j and $\varepsilon = 10^{-16}$ is to avoid dividing by zero.

With the above defined weights, the proposed WNNM algorithm in subsection 2.2.3 can be directly used to solve the model in (9). However, there is still one problem remaining, that is, the singular values $\sigma_i(X_j)$ are not available. We assume that the noise energy is evenly distributed over each subspace spanned by the basis pair of U and V,

and then the initial $\sigma_i(X_j)$ can be estimated as

$$\hat{\sigma}_i(X_j) = \sqrt{\max(\sigma_i^2(Y_j) - n\sigma_n^2, 0)},$$

where $\sigma_i(Y_j)$ is the i-th singular value of Y_j. Note that the obtained weights $w_{i=1,\dots,n}$ are guaranteed to be in a non-descending order since $\hat{\sigma}_i(X_j)$ are always sorted in a non-ascending order. By applying the above procedures to each patch and aggregating all patches together, the image x can be reconstructed. In practice, we can run several more rounds of those procedures to enhance the denoising outputs. The whole denoising algorithm is summarized in Algorithm 1.

Algorithm 1 Image Denoising by WNNM

Input: Noisy image y
1: Initialize $\hat{x}^{(0)} = y, y^{(0)} = y$
2: **for** $k=1:K$ **do**
3:    Iterative regularization $y^{(k)} = \hat{x}^{(k-1)} + \delta(y - \hat{y}^{(k-1)})$
4:    **for** each patch y_j in $y^{(k)}$ **do**
5:       Find similar patch group Y_j
6:       Estimate weight vector w
7:       Singular value decomposition $[U, \Sigma, V] = SVD(Y_j)$
8:       Get the estimation: $\hat{X}_j = US_w(\Sigma)V^T$
9:    **end for**
10:   Aggregate X_j to form the clean image $\hat{x}^{(k)}$
11: **end for**
Output: Clean image $\hat{x}^{(K)}$

---

4. Experiments

We compare the proposed WNNM based image denoising algorithm with several state-of-the-art denoising methods, including BM3D [7], EPLL [31], LSSC [22], NCSR [10] and SAIST [9]. The baseline NNM algorithm is also compared. All the competing methods exploit the image nonlocal redundancies. In subsection 4.1, we discuss the parameter settings in the WNNM denoising algorithm; in subsection 4.2, we evaluate WNNM and its competing methods on 20 widely used test images.

4.1. Parameter Setting

There are several parameters (δ, c, K and patch size) in the proposed algorithm. For all noise levels, the iterative regularization parameter δ and the parameter c are fixed to 0.1 and 2.8, respectively. Iteration number K and patch size are set based on noise level. For higher noise level, we need to choose bigger patches and run more times the iteration. By experience, we set patch size to 6×6, 7×7, 8×8 and 9×9 for $\sigma_n \leq 20$, $20 < \sigma_n \leq 40$, $40 < \sigma_n \leq 60$ and $60 < \sigma_n$, respectively. K is set to 8, 12, 14, and 14 respectively, on these noise levels.

For NNM, we use the same parameters as WNNM except for the uniform weight $\sqrt{n}\sigma_n$. The source codes of the

Figure 1. The 20 test images.

competing methods are obtained from the original authors, and we use the default parameters.

4.2. Experimental Results on 20 Test Images

We evaluate the competing methods on 20 widely used test images, whose scenes are shown in Fig. 1. Zero mean additive white Gaussian noises with variance σ_n^2 are added to those test images to generate the noisy observations. Due to page limit, we show the results on four noise levels, ranging from low noise level $\sigma_n = 10$, to medium noise levels $\sigma_n = 30$ and 50, and to strong noise level $\sigma_n = 100$. Results on more noise levels can be found in the supplementary material. The PSNR results by the competing denoising methods are shown in Table 1. The highest PSNR result for each image and on each noise level is highlighted in bold. We have the following observations. First, the proposed WNNM achieves the highest PSNR in almost every case. It achieves 1.3dB-2dB improvement over the NNM method in average and outperforms the benchmark BM3D method by 0.3dB-0.45dB in average (up to 1.16dB on image *Leaves* with noise level $\sigma_n = 10$) consistently on all the four noise levels. Such an improvement is notable since few methods can surpass BM3D more than 0.3dB in average [18, 17]. Second, some methods such as LSSC and NCSR can outperform BM3D a little when the noise level is low, but their PSNR indices become almost the same as, or lower than, those of BM3D with the increase of noise level. This shows that the proposed WNNM method is more robust to noise strength than other methods.

In Fig. 2 and Fig. 3, we compare the visual quality of the denoised images by the competing algorithms (more visual comparison results can be found in the supplementary material). Fig. 2 demonstrates that the proposed WNNM reconstructs more image details from the noisy observation. Compared with WNNM, methods LSSC, NCSR and SAIST over-smooth more the textures in the sands area of image *Boats*, and methods BM3D and EPLL generate more artifacts. More interestingly, as can be seen in the highlighted window, the proposed WNNM can still well reconstruct the tiny masts of the boat, while the masts are almost disappeared in the reconstructed images by other methods. Fig. 3 shows an example with strong noise. It is obvious that WNNM generates much less artifacts and preserves much better the image edge structures than other competing methods. In summary, WNNM shows strong denoising capability, producing visually much more pleasant denoising outputs while having higher PSNR indices.

5. Conclusion

As a significant extension of the nuclear norm minimization problem, the weighted nuclear norm minimization (WNNM) was studied in this paper. We showed that, when the weights are in a non-ascending order, WNNM is still convex and we presented the analytical optimal solution; when the weights are in an arbitrary order, we presented an iterative algorithm to solve it; when the weights are in a non-descending order, we proved that the iterative algorithm can result in an analytical fixed point solution, which can be efficiently computed. We then applied the proposed WNNM algorithm to image denoising. The experimental results showed that WNNM can not only lead to visible PSNR improvements over state-of-the-art methods such as BM3D, but also preserve much better the image local structures and generate much less visual artifacts. It can be expected that WNNM will have more successful applications in computer vision problems.

References

[1] A. Buades, B. Coll, and J.-M. Morel. A non-local algorithm for image denoising. In *CVPR*, 2005.

[2] A. M. Buchanan and A. W. Fitzgibbon. Damped newton algorithms for matrix factorization with missing data. In *CVPR*, 2005.

[3] J.-F. Cai, E. J. Candès, and Z. Shen. A singular value thresholding algorithm for matrix completion. *SIAM Journal on Optimization*, 20(4):1956–1982, 2010.

[4] E. J. Candès, X. Li, Y. Ma, and J. Wright. Robust principal component analysis? *JACM*, 58(3):11, 2011.

[5] E. J. Candès and Y. Plan. Matrix completion with noise. In *Proceedings of the IEEE*, 2010.

[6] E. J. Candès and B. Recht. Exact matrix completion via convex optimization. *Foundations of Computational mathematics*, 9(6):717–772, 2009.

[7] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian. Image denoising by sparse 3-d transform-domain collaborative filtering. *TIP*, 16(8):2080–2095, 2007.

[8] F. De La Torre and M. J. Black. A framework for robust subspace learning. *IJCV*, 54(1-3):117–142, 2003.

[9] W. Dong, G. Shi, and X. Li. Nonlocal image restoration with bilateral variance estimation: a low-rank approach. *TIP*, 22(2):700–711, 2013.

[10] W. Dong, L. Zhang, and G. Shi. Centralized sparse representation for image restoration. In *ICCV*, 2011.

[11] D. L. Donoho, M. Gavish, and A. Montanari. The phase transition of matrix recovery from gaussian measurements matches the minimax mse of matrix denoising. In *PNAS*, 2013.

[12] M. Elad and M. Aharon. Image denoising via sparse and redundant representations over learned dictionaries. *TIP*, 15(12):3736–3745, 2006.

[13] A. Eriksson and A. Van Den Hengel. Efficient computation of robust low-rank matrix approximations in the presence of missing data using the l_1 norm. In *CVPR*, 2010.

(i) Ground truth   (j) Noisy image (PSNR: 14.16dB)   (k) BM3D (PSNR: 26.78dB)   (l) EPLL (PSNR: 26.65dB)

(m)LSSC (PSNR: 26.77dB)   (n) NCSR (PSNR: 26.66dB)   (o) SAIST (PSNR: 26.63dB)   (p) WNNM (PSNR: 26.98dB)
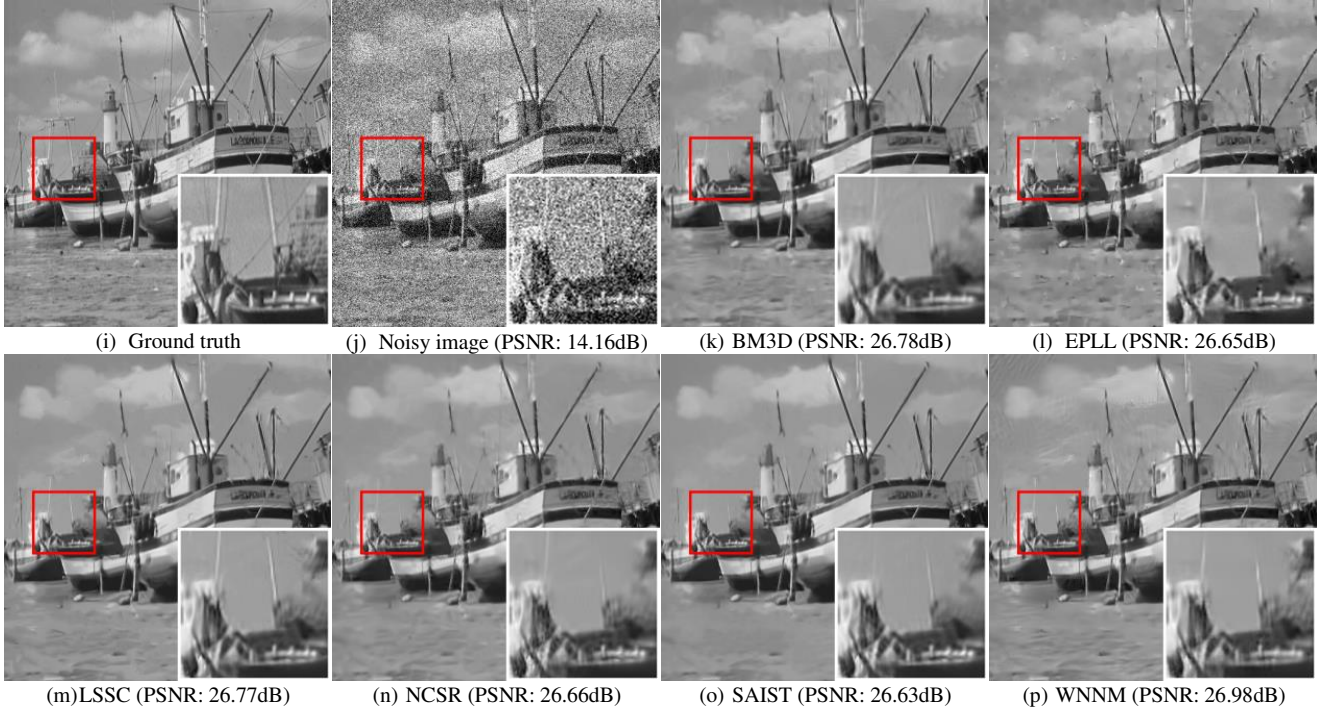
Figure 2. Denoising results on image *Boats* by different methods (noise level $\sigma_n = 50$).

(a) Ground truth   (b) Noisy image (PSNR: 8.10dB)   (c) BM3D (PSNR: 22.52dB)   (d) EPLL (PSNR: 22.23dB)

(e) LSSC (PSNR: 22.24dB)   (f) NCSR (PSNR: 22.11dB)   (g) SAIST (PSNR: 22.61dB)   (h) WNNM (PSNR: 22.91dB)
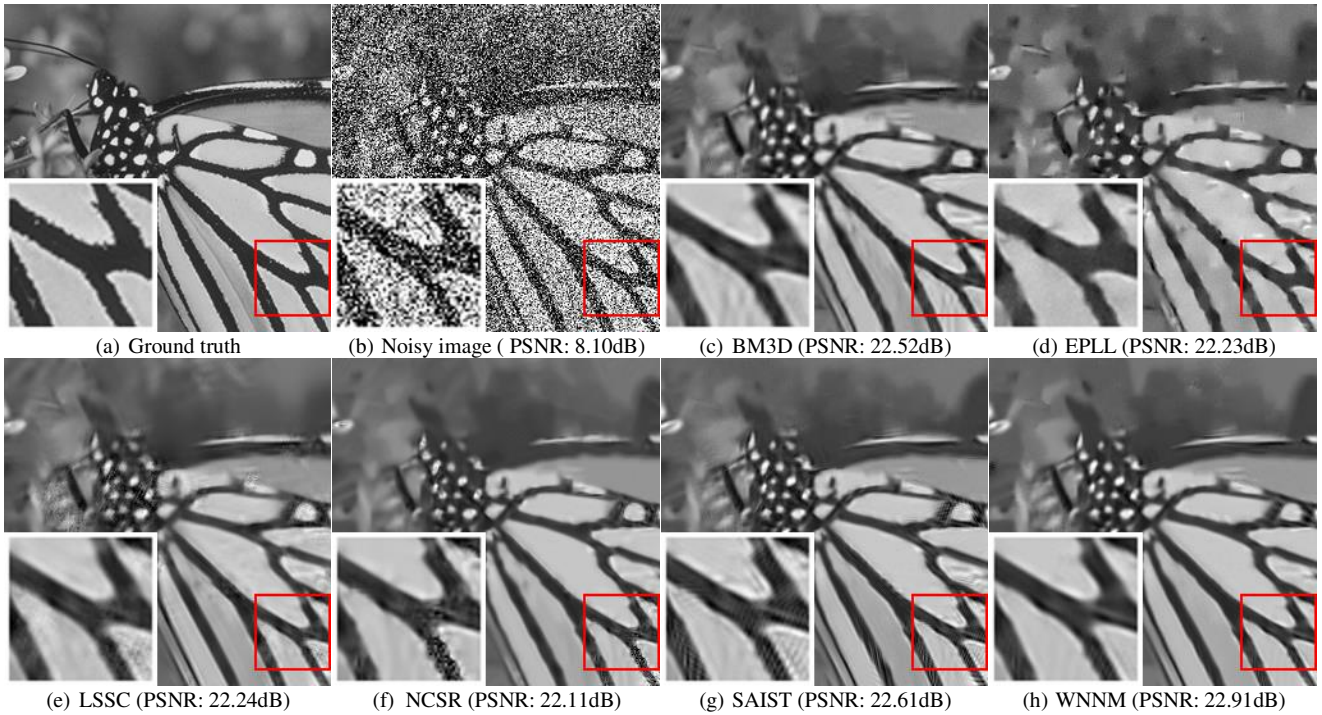
Figure 3. Denoising results on image *Monarch* by different methods (noise level $\sigma_n = 100$).

[14] M. Fazel, H. Hindi, and S. P. Boyd. A rank minimization heuristic with application to minimum order system approximation. In *ACC*, 2001.

[15] H. Ji, C. Liu, Z. Shen, and Y. Xu. Robust video denoising using low rank matrix completion. In *CVPR*, 2010.

[16] Q. Ke and T. Kanade. Robust l_1 norm factorization in the presence of outliers and missing data by alternative convex programming. In *CVPR*, 2005.

Table 1. Denoising results (PSNR) by different methods.

	σ_n=10							σ_n=30						
	NNM	BM3D	EPLL	LSSC	NCSR	SAIST	WNNM	NNM	BM3D	EPLL	LSSC	NCSR	SAIST	WNNM
C.Man	32.87	34.18	34.02	34.24	34.18	34.30	**34.44**	27.43	28.64	28.36	28.63	28.59	28.36	**28.80**
House	35.97	36.71	35.75	**36.95**	36.80	36.66	**36.95**	30.99	32.09	31.23	32.41	32.07	32.30	**32.52**
Peppers	33.77	34.68	34.54	34.80	34.68	34.82	**34.95**	28.11	29.28	29.16	29.25	29.10	29.24	**29.49**
Montage	36.09	37.35	36.49	37.26	37.17	37.46	**37.84**	29.28	31.38	30.17	31.10	30.92	31.06	**31.65**
Leaves	33.55	34.04	33.29	34.52	34.53	34.92	**35.20**	26.81	27.81	27.18	27.65	28.14	28.29	**28.60**
StarFish	32.62	33.30	33.29	33.74	33.65	33.72	**33.99**	26.62	27.65	27.52	27.70	27.78	27.92	**28.08**
Monarch	33.54	34.12	34.27	34.44	34.51	34.76	**35.03**	27.44	28.36	28.35	28.20	28.46	28.65	**28.92**
Airplane	32.19	33.33	33.39	33.51	33.40	33.43	**33.64**	26.53	27.56	27.67	27.53	27.53	27.66	**27.83**
Paint	33.13	34.00	34.01	34.35	34.15	34.28	**34.50**	27.02	28.29	28.33	28.29	28.10	28.44	**28.58**
J.Bean	37.52	37.91	37.63	38.69	38.31	38.37	**38.93**	31.03	31.97	31.56	32.39	32.13	32.14	**32.46**
Fence	32.62	33.50	32.89	33.60	33.65	33.76	**33.93**	27.19	28.19	27.23	28.16	28.23	28.26	**28.56**
Parrot	32.54	33.57	33.58	33.62	33.56	33.66	**33.81**	27.26	28.12	28.07	27.99	28.07	28.12	**28.33**
Lena	35.19	35.93	35.58	35.83	35.85	35.90	**36.03**	30.15	31.26	30.79	31.18	31.06	31.27	**31.43**
Barbara	34.40	34.98	33.61	34.98	35.00	35.24	**35.51**	28.59	29.81	27.57	29.60	29.62	30.14	**30.31**
Boat	33.05	33.92	33.66	34.01	33.91	33.91	**34.09**	27.82	29.12	28.89	29.06	28.94	28.98	**29.24**
Hill	32.89	33.62	33.48	33.66	33.69	33.65	**33.79**	28.11	29.16	28.90	29.09	28.97	29.06	**29.25**
F.print	31.38	32.46	32.12	32.57	32.68	32.69	**32.82**	25.84	26.83	26.19	26.68	26.92	26.95	**26.99**
Man	32.99	33.98	33.97	34.10	34.05	34.12	**34.23**	27.87	28.86	28.83	28.87	28.78	28.81	**29.00**
Couple	32.97	34.04	33.85	34.01	34.00	33.96	**34.14**	27.36	28.87	28.62	28.77	8.57	28.72	**28.98**
Straw	29.84	30.89	30.74	31.25	31.35	31.49	**31.62**	23.52	24.84	24.64	24.99	25.00	25.23	**25.27**
AVE.	33.462	34.326	34.008	34.507	34.456	34.555	**34.772**	27.753	28.905	28.463	28.877	28.849	28.980	**29.214**
	σ_n=50							σ_n=100						
C-Man	24.88	26.12	26.02	26.35	26.14	26.15	**26.42**	21.49	23.07	22.86	23.15	22.93	23.09	**23.36**
House	27.84	29.69	28.76	29.99	29.62	30.17	**30.32**	23.65	25.87	25.19	25.71	25.56	26.53	**26.68**
Peppers	25.29	26.68	26.63	26.79	26.82	26.73	**26.91**	21.24	23.39	23.08	23.20	22.84	23.32	**23.46**
Montage	26.04	27.9	27.17	28.10	27.84	28.0	**28.27**	21.70	23.89	23.42	23.77	23.74	23.98	**24.16**
Leaves	23.36	24.68	24.38	24.81	25.04	25.25	**25.47**	18.73	20.91	20.25	20.58	20.86	21.40	**21.57**
Starfish	23.83	25.04	25.04	25.12	25.07	25.29	**25.44**	20.58	22.10	21.92	21.77	21.91	22.10	**22.22**
Mornar.	24.46	25.82	25.78	25.88	25.73	26.10	**26.32**	20.22	22.52	22.23	22.24	22.11	22.61	**22.95**
Plane	23.97	25.10	25.24	25.25	24.93	25.34	**25.43**	20.73	22.11	22.02	21.69	21.83	22.27	**22.55**
Paint	24.19	25.67	25.77	25.59	25.37	25.77	**25.98**	21.02	22.51	22.50	22.14	22.11	22.42	**22.74**
J.Bean	27.96	29.26	28.75	29.42	29.29	29.32	**29.62**	23.79	25.80	25.17	25.64	25.66	25.82	**26.04**
Fence	24.59	25.92	24.58	25.87	25.78	26.00	**26.43**	21.23	22.92	21.11	22.71	22.23	22.98	**23.37**
Parrot	24.87	25.90	25.84	25.82	25.71	25.95	**26.09**	21.38	22.96	22.71	22.79	22.53	23.04	**23.19**
Lena	27.74	29.05	28.42	28.95	28.90	29.01	**29.24**	24.41	25.95	25.30	25.96	25.71	25.93	**26.20**
Barbara	25.75	27.23	24.82	27.03	26.99	27.51	**27.79**	22.14	23.62	22.14	23.54	23.20	24.07	**24.37**
Boat	25.39	26.78	26.65	26.77	26.66	26.63	**26.97**	22.48	23.97	23.71	23.87	23.68	23.80	**24.10**
Hill	25.94	27.19	26.96	27.14	26.99	27.04	**27.34**	23.32	24.58	24.43	24.47	24.36	24.29	**24.75**
F.print	23.37	24.53	23.59	24.26	24.48	24.52	**24.67**	20.01	21.61	19.85	21.30	21.39	21.62	**21.81**
Man	25.66	26.81	26.72	26.72	26.67	26.68	**26.94**	22.88	24.22	24.07	23.98	24.02	24.01	**24.36**
Couple	24.84	26.46	26.24	26.35	26.19	26.30	**26.65**	22.07	23.51	23.32	23.27	23.15	23.21	**23.55**
Straw	20.99	22.29	21.93	22.51	22.30	22.65	**22.74**	18.33	19.43	18.84	19.43	19.10	19.42	**19.67**
AVE.	25.048	26.406	25.965	26.436	26.326	26.521	**26.752**	21.570	23.247	22.706	23.061	22.996	23.296	**23.555**

[17] A. Levin and B. Nadler. Natural image denoising: Optimality and inherent bounds. In *CVPR*, 2011.

[18] A. Levin, B. Nadler, F. Durand, and W. T. Freeman. Patch complexity, finite pixel correlations and optimal denoising. In *ECCV*. 2012.

[19] Z. Lin, R. Liu, and Z. Su. Linearized alternating direction method with adaptive penalty for low-rank representation. In *NIPS*, 2011.

[20] G. Liu, Z. Lin, S. Yan, J. Sun, Y. Yu, and Y. Ma. Robust subspace segmentation by low-rank representation. In *ICML*, 2010.

[21] R. Liu, Z. Lin, F. De la Torre, and Z. Su. Fixed-rank representation for unsupervised visual learning. In *CVPR*, 2012.

[22] J. Mairal, F. Bach, J. Ponce, G. Sapiro, and A. Zisserman. Non-local sparse models for image restoration. In *ICCV*, 2009.

[23] Y. Mu, J. Dong, X. Yuan, and S. Yan. Accelerated low-rank visual recovery by random projection. In *CVPR*, 2011.

[24] R. Salakhutdinov and N. Srebro. Collaborative filtering in a non-uniform world: Learning with the weighted trace norm. In *NIPS*, 2010.

[25] N. Srebro, T. Jaakkola, et al. Weighted low-rank approximations. In *ICML*, 2003.

[26] S. Wang, L. Zhang, and L. Y. Nonlocal spectral prior model for low-level vision. In *ACCV*, 2012.

[27] J. Wright, Y. Peng, Y. Ma, A. Ganesh, and S. Rao. Robust principal component analysis: Exact recovery of corrupted low-rank matrices via convex optimization. In *NIPS*, 2009.

[28] D. Zhang, Y. Hu, J. Ye, X. Li, and X. He. Matrix completion by truncated nuclear norm regularization. In *CVPR*, 2012.

[29] Z. Zhang, A. Ganesh, X. Liang, and Y. Ma. Tilt: transform invariant low-rank textures. *IJCV*, 99(1):1–24, 2012.

[30] Y. Zheng, G. Liu, S. Sugimoto, S. Yan, and M. Okutomi. Practical low-rank matrix approximation under robust l_1 norm. In *CVPR*, 2012.

[31] D. Zoran and Y. Weiss. From learning models of natural image patches to whole image restoration. In *ICCV*, 2011.